An Exploratory Qualitative and Quantitative Study into the Effectiveness of Digital Games as a Tool to Enhance the Learning of Introductory Programming

Wong Seng Yue[1], Wong Lai Wan[2]

[1]*School of Computing and Information Technology Taylor's University Malaysia*
[2]*School of Information Technology UCSI University Malaysia*

Abstract

This study presents the details of an initial study that is conducted to identify the features and characteristics of using games to improve the ability of students to code. The objective of this study is to examine the quality of a game meant for learning programming concepts. Participants in the exploratory study were 42 students from a local higher education institution. Subjective data were collected with pre- and post-use questionnaires whereas objective data was gathered with interviews, observations and in the form of in-game progress. The analysis of collected data uncovered: (1) to what extent students with different background knowledge differ in their perception of programming and attitudes towards using games for educational purposes, and (2) which quality is relevant for the assessment of games designed for learning programming. All students are neutral in their past programming experience and neutral receptive towards programming prior to the digital game workshop. After the workshop, they feel the digital game has enhanced their understanding in basic programming concepts (60.9% of first semester students and 57.9% of second semester students) and ability to conceptualize the solution to a particular problem (69.6% of first semester students and 73.7% of second semester students). Moreover, the digital game has helped to improve their perceptions on programming and dispel their negative thoughts on programming.

1. Introduction

There is a growing trend to introduce the fundamentals of programming into various groups of school going children as well as those who are in fields of studies other than computer science. Many studies have been conducted and ample evidence of the benefits of learning to code have been documented in various leves of education ranging from primary to tertiary levels. In a recent definition for K-12, computational thinking is described as a problem-solving process that includes formulating problems, logically organizing and analyzing data, representing data through abstractions, and automating solutions [1]. Recent studies confirmed that Computer Games Programming is motivating [2-3] and increases girls' perceived computer skills and perceived support for computing [4-5].

Additionally, digital games are also increasingly explored and adopted as a means to complement teaching and learning in various fields of studies, and also at different levels of education covering extensively primary to university level. Our review of literature uncovered many studies on the efficacy of games-based education that have been conducted and the results have been encouraging. The idea of leveraging on the richness of games based educational means in particular to learn coding is not new. To the best of our knowledge, there have been many studies that were conducted to review the effectiveness of using games for teaching and learning programming from different angles and perspective of learning coding. Our list of literature review shows amongst the others, include (1) Interactive Story-telling using Alice and Scratch [6-7], (2) Using simple maze-like problem solving games that appeals to both young learners as well as adults to learn popular languages like JavaScript and Python as in Code.org [8] and (3) Debugging-first approach to learning programming concepts via Gidget [9].

Learning to code has always been considered a very difficult task– both for the teacher and learner. Traditionally, practical exercises and text based consoles were the means that teachers employ to help students consolidate knowledge acquired. Students have to demonstrate proficiency in a certain

topic by applying theoretical knowledge to particular problems and keeping track of students' progression is challenging when we consider the class size and that indicators of students' learning is limited to mainly the result of work submissions. These teaching methods are considered to be inadequate because it is not an accurate indicator of students' skills to problem solve thereby further pointing to the suggestion that the underlying mechanisms of performance gains might be different between traditional learning methods and game training [10-11].

This can probably be attributed to the change in demands of the young generations in terms of learning coupled with a constant need for higher order engaging tasks. In this 21st century, the new group of younger and emerging generations need to be educated in a similar manner that stimulates high order thinking skills and provided with problem solving opportunities within a communicative mode that promotes a unique kind of social relationships [12]. It has often been thought that logical reasoning and algorithmic thinking are essential to succeed in learning programming. It has been demonstrated that children who played games benefitted and showed improvement in their understanding of programming concepts like sequence, iteration and decision making which are all important in learning how to code which in turn enhances their ability to problem solve. Similar results suggesting that playing games can lead to gains in engagement, logical and analytical skills, and improvement in attitude towards programming have also been reported. All the studies above suggest positively that learners benefit from playing games as compared to those who do not.

In this study, we used a Pacman game to investigate the effectiveness of digital games in enhancing the learning of introductory programming via an exploratory qualitative and quantitative study. Literature reviews have been done regrading programming concepts learning and utilizing digital games in teching and learning of programming. Then, we present and discuss the results and findings of the exploratory study. We hope the significant findings from the sudy can contribute to the future studies in programming concepts learning.

2. Literature review

Many studies have been conducted to investigate the effectiveness of using games in teaching and learning of various fields of study, theatre, economic, computer language implementation course, cognitive abilities and aritmethic performances, science and entrepreneurship education [11, 13-14]. Cheng [12]

proposes an online competitive game-based learning containing learning points, competition mechanism, training room mechanism, questioning & answering mechanism, tips mechanism, and feedback mechanism. Particularly, the competition mechanism can effectively enhance the students' learning interests, performance, and willingness to learn continuously. Table 1 presents a number of studies that have been conducted to investigate the outcome of a change in programming environments to the medium used for teaching and its evidence where available.

All of the reports presented in the earlier sections show that there is much to be gained from inclusion of gaming into curriculum and knowledge of programming can be transferred from various angles and possibly by taking into consideration various learning styles of students. Typical factors that gauge a student's ability on programming include problem-solving, self-regulated learning, and computational thinking. Success of any adaptation can be factored around these:

- Knowledge of Syntax (requiring memory work) and is a determinant in student learning of programming;
- Clear instruction and Step-by-step tutorial showing breakdown of individual steps to be performed;
- Representation of basic concepts should be in a tangible and a visible way;
- Interactive Story-telling mode ;
- Adaptivity (comfort and experience) has the potential to significantly short completion time – there is a noted difference between GBL based and paper based adaptivity.

Hence in order for a learner to be considered as having successfully understood and learnt a programming course, the learner typically should be able to transform a formulation of a conceptual design model to an implementation version that is eventually "executable". The creation of such a conceptual model requires problem solving skills which are eventually translated using syntax by representing concepts.

This process requires:
1. To be able to construct a conceptual representation of a real world problem based on
    a. Ability to represent basic concepts
    b. Express the concepts in a sequence , that is
    c. Close to that of a process that is gained through interactive story-telling mode.
2. Transformation of conceptual representation into an implementation requires:
    a. Knowledge of syntax

b. Computational thinking to problem solve

Hence, the issue that we wish to bring into attention is at the transition from 1 into 2. How does game play which corresponds so closely to the conceptual logical train of solution that is crucial to programming solution can lead to its actual translation in terms of syntax? In other words, to what extent does game play add value to the traditional method of learning how to code which typically begins by introduction of syntaxes and different constructs that needs to be placed together in order to form the "solution" of a program? Another area that is worth mentioning is the paradigm of the programming language itself. We feel that many game based learning tools that are available largely subscribe to the concept of object identification and separation. To us, this corresponds closely to an object oriented programming paradigm but whether the same can be said for learning of non-object based procedural languages remains an interesting area to be explored. But it must be said that for these two seemingly different paradigms, there is still a similar process in the transformation of an initial concept of solution to the coding to the implementation of an executable version.

As with regards to the game itself, what are the features and characteristics of such a game that is required and to what extent do these factors add on to bring upon the engagement and motivation that are widely considered as important aspects of actual learning? Perhaps yet another area that is of concern is whether learners' durability can be sustained and measured as this usually requires a constant change in the mode of game play and story line. While many studies have proven gaming and its effectiveness in teaching programming, what are the actual learning gains in papers and written exercises? This is what our proposed study tries to uncover.

Table 1. Games used for teaching and learning of programming

Author(s)	Games / Tools	Area of study	Evaluation results
Ouahbi et al. [15] (2014)	Scratch game	Effectiveness of learning basic programming concepts by creating games with Scratch game environment	65% students who have experienced with Scratch game environment will consider continuing their studies in programming.
Mohd Nor et al. [16] (2012)	Instructional design and learning theory	The possibilities of using Gagne's theory and instructional design on teaching and learning computer programming	The findings indicated Gagne theory can be used to provide and assist students in computer programming learning.
Kalelioğlu[8] (2015)	Code.org	Teaching programming skills to K-12 students via Code.org	Students developed a positive attitude towards programming, female students showed that they were as successful as male students and programming can be part of their future plans.
Ahmad Zamzuri and SitiRosminah[17] (2013)	Hardware-Software interfacing visualization kit	Usability and user satisfaction of hardware-software interfacing visualization kit for novice learning programming	The developed tool seems very helpful and useful for beginners in understanding hardware-software interfacing programming.
Wang et al. [18] (2012)	Peer code review model	Implementation of assessment programming language learning based on peer code review model	The assessment approach demonstrates high practical values in assessing students' learning outcomes in programming languages.

3. Methodology

This research will investigate the effects of game-based learning (GBL) approach by using a digital game over a period of two hour workshop specifically studying the possible improvements on understanding of fundamental programming concepts, on ability to solve a programmatic problem and on their interest in the subject of computer programming. This study will involve two groups of students, 19 second semester students Foundation of Computing and 23 first semester students from

Foundation of Computing. Second semester students mostly have experienced and are exposed to at least one programming subject. On the other hand, first semester students are fresh graduated students from secondary schools and have neither experience nor exposure to programming subject in a formal taught course. The instrument was modified from a previous study [19].

These two groups of students will undergo a two hours programming workshop that requires them to play a digital game. Two questionnaires were used to address the aim of the research. First part of the questionnaire (for pre-workshop) consists of open and close ended questions which will be distributed before the workshop to determine participants' demographic, programming background and attitudes toward programming. Four items were rated on a 5-point Likert scale ranging from 1 to 5 (1-Strongly Dislike; 2-Dislike; 3-Neutral; 4-Like; 5-Strongly Like). This questionnaire was devised to determine participants:

- Gender
- Age group
- Familiarity with programming language and the level of competency
- Opinion on how to learn programming
- Past programming experience
- Receptiveness towards programming
- Issues of learning programming
- Thoughts on learning programming

The second part of the questionnaires (for post-workshop) consists of open and close ended questions which will be distributed after the workshop to determine participants' programming experiences of, using digital game as a tool to enhance the learning of introductory programming concepts, as well as effectiveness of understanding introductory programming concepts. After the workshop, this part questionnaire also determined the effectiveness of the digital game in learning programming. Six items were rated on a 5-point Likert scale ranging from 1to 5 (1-Strongly Dislike; 2-Dislike; 3-Neutral; 4-Like; 5-Strongly Like). This questionnaire was devised to determine the participants:

- Programming experience during workshop
- Difficulty experienced in regards to solving the game problem
- Thoughts on use of digital games to enhance understanding of basic programming concepts
- Thoughts on digital games to enhance the ability to conceptualize the solution to a particular problem

- Thoughts on effectiveness of digital games in helping problem solving
- Opinions on what they like and dislike in the game
- Thoughts on effectiveness of understanding introductory programming concepts via different elements, playability, multimedia adequacy, sensory immersion, ease of use, challenge, enjoyment, learnability, interaction, context adequacy and pedagogical utility.
- Thoughts on the effect of digital games on perception of programming

4. Findings and results analysis

As presented in the methodology section, both questionnaires (pre- and post-workshop) have been subject to quantitative and qualitative analysis and descriptive statistics, figures and the tabulation of data have been used. This exploratory study was designed to be an initial investigation on effects of a digital game, GBL approach to enhance programming learning. Responses of open ended questions were analyzed by identifying the trends in the received replies and by grouping these information into groups. Quotes have been used to illustrate examples of identified trends.

There were 23 students (19 Male, 4 Female) from first semester of Foundation in Computing at Taylor's University participating in the study. These participants were in age of tertiary education level and mostly (74%) have not learnt any computer languages prior to the digital game workshop. However, 19 students (15 Male, 4 Female) from second semester of Foundation in Computing at Taylor's University in the age of tertiary education level and all participants have learnt at least one computer language prior to the workshop. Table II showns the comparison of programming languages experience between the two groups of participants according to their level of competency: Beginner, Competence and Advanced level. Of the 25 participants (first and second semester students), 22 students (88%) had learnt Java programming and 20 students are at beginner level. Besides that, there are 15 participants also have learnt HTML and 9 participants have learnt C/ C++/ C# programming. Of six first semester students with programming experience, 50% of them are self-taught and another 50% students learnt programming due to part of course. Most of the participants (73.7%) have learnt programming via part of a course for second semester students and three participants use both methods to learn programming.

Table 3 shows the responses of first semester participants for their past programming experience

and receptiveness towards programming. On the other hand, responses of second semester participants are shown in Table 4. The differences in their responses among second semester students are not significant, but most of the first semester students rated their past programming experience as neutral. Both first and second semester students also rated neutral in their receptiveness towards programming.

Although most of the participants, from both first semester and second semester participants, rated neutral in their past programming experience, however, they perceived that learning programming is good and indicated they would like to learn programming in their spare time if they were given support. Table 5 presents the responses of the first semester participants on this. Second semester

participants' responses are presented in Table 6. Of 23 first semester students, 91.3% students highlighted like and strongly like in considering learning programming in their spare time and 91.3% students chosed like and strongly believed that learning programming is good. 89.5% students of 19 second semester students indicated that they like and strongly like to consider learning programming in their spare time and believed it is good to learn programming. Amongst the issues highlighted by the participants pertaining to barrier in learning programming include: difficult to learn, hard to understand, confused by the loop strategy and easy to lose track.

Table 2. Programming languages that have learnt by participants had experienced according to their level of competency

Participants	First semester students				Second semester students			
Programming Language	Beginner	Competence	Advanced	Total number of participants with experience	Beginner	Competence	Advanced	Total number of participants with experience
Java	3	0	0	3	17	2	0	19
HTML	2	1	0	3	9	3	0	12
C / C++/C#	1	0	0	1	5	3	0	8
Python	0	0	0	0	1	0	0	1
VB.Net	0	0	0	0	1	0	0	1
Pascal	0	0	0	0	1	0	0	1
Basic	0	0	0	0	0	0	1	1
JavaScript	0	0	0	0	0	1	0	1
CSS	1	0	0	1	1	0	0	1
Ruby	1	0	0	1	0	0	0	0
Swift	0	0	0	0	1	0	0	1

Table 3. First semester participants responses on their past programming experience and receptive towards programming

	"How would you describe your past programming experience?"				
	Strongly Dislike	**Dislike**	**Neutral**	**Like**	**Strongly Like**
Number of responses	0	0	18	3	2
%	0	0	78.3	13	8.7
	"How receptive are you towards programming?"				
	Strongly Dislike	**Dislike**	**Neutral**	**Like**	**Strongly Like**
Number of responses	0	0	12	7	4
%	0	0	52.2	3.1	1.7

Table 4. Second semester participants responses on their past programming experience and receptive towards programming

	Strongly Dislike	Dislike	Neutral	Like	Strongly Like
"How would you describe your past programming experience?"					
Number of responses	0	0	6	6	7
%	0	0	31.6	31.6	36.8
"How receptive are you towards programming?"					
Number of responses	0	1	6	7	5
%	0	5.3	31.6	36.8	26.3

Table 5. First semester participants'thoughts on learning programming

	Strongly Dislike	Dislike	Neutral	Like	Strongly Like
"Would you consider learning programming in your spare time if you are given support?"					
Number of responses	0	0	2	8	13
%	0	0	8.7	34.8	56.5
"Do you believe it is good to learn programming?"					
Number of responses	0	0	2	9	12
%	0	0	8.7	39.1	52.2

Table 6. Second semester participants' thoughts on learning programming

	Strongly Dislike	Dislike	Neutral	Like	Strongly Like
"Would you consider learning programming in your spare time if you are given support?"					
Number of responses	0	0	2	9	8
%	0	0	10.5	47.4	42.1
"Do you believe it is good to learn programming?"					
Number of responses	0	0	2	3	14
%	0	0	10.5	15.8	73.7

To conclude the findings of the pre-workshop, we found that first semester tertiary education group students mostly have no experience with programming, neutral in their past programming experience and receptive towards programming, but like and strongly like to learn programming in their spare time and believe that learning programming is good. All second semester students have learnt programming (Java, HTML and C/C++/C#) prior the workshop due to part of their courses. They are neutral in their past programming experience and neutral receptive towards programming too. However, most of them still consider learning programming in their spare time and believe it is good to learn programming.

After two hours workshop, we review participants' opinions of effects of digital games to enhance their understanding on basic programming concepts, ability to conceptualize the solution to a particular problem and effectiveness of digital game in helping to first identify a particular pattern to problem solving and eventually implement it in a language that they are familiar with. Table 7 presented the responses of the first semester participants on effects of digital games. Table 8 shows the second semester participants' opinions on effects digital games. Both groups of participants gave positive responses on that. 60.9% of first semester participants and 57.9% of second semester participants like and strongly felt that the digital game has enhanced their understanding in basic programming concepts. Amongst the groups, 69.6 % of first semester participants and 73.7 % of second semester participants chosed like and strongly like in the question on whether the digital game has enhanced their ability to conceptualize the solution to a particular problem. 69.5 % of first semester participants and 63.2 % of second semester participants also like and strongly like that digital game is effective in helping them to first identify a particular pattern to problem solving and eventually implement is in a language that they are familiar with.

Furthermore, participants also rated the effectiveness elements for understanding introductory programming concepts. Table 9 shows the mean score of participants' likeliness on effectiveness elements of their understanding

introductory programming concepts. All results shown were above 3.000 except sensory immersion element from the perspective of second semester participants. This means that almost all first semester participants were neutral for the effectiveness elements (playability, multimedia adequacy, sensory immersion, ease of use, enjoyment, interaction, context adequacy and pedagogical utility) for understanding introductory programming concepts. They like the challenges and learnability elements of the digital game, which scored a mean rating of 3.522 and 3.609 respectively. However, second semester participants were neutral in multimedia adequacy, ease of use, enjoyment, interaction, context adequacy and pedagogical utility elements of digital game, which shows a mean between 3.000 to 3.500. They liked the playability, challenge and learnability elements of digital game.

Besides that, participants also suggested their like and dislike features of digital game. Most of them liked the digital game's ease of use, simple, problem solving, challenge, fun and enjoyment features. However, participants mostly dislike the digital game's slow moving character, buggy, lack of instruction, lack of 'forward' or 'arrow' buttons, boring, confuse, no graphics and repetitive tasks.

Lastly, we also viewed the responses of participants on how the digital game improved their perception of programming, dispels negative thoughts on programming and considering programming learning in their spare time. Thoughts on good to learn programming also have been recorded from the participants. Table 10 shows the responses of first semester participants on their perception of programming after they are exposed to the digital game.

Meanwhile, Table 11 presents the responses of second semester participants on their perception of programming. Overall responses received on improvement in perception of programming for first semester participants are 56.5% and second semester participants are 68.4%. At least half of them have improved their perception on programming and digital game also dispelled their negative thoughts, which is 65.2% from first semester participants and 52.6% from second semester participants. After the workshop, most of the participants (87% first semester participants and 68.4% of second semester participants) indicated a continued willingness in considering learning programming in their spare time. 87% of first semester participants and 78.9% of second semester participants maintained their belief that it is good to learn programming after being exposed to digital game workshop.

Table 7. First semester participants' opinions on effect of digital game in enhancing understanding in basic programming concepts and conceptualize the solution to a particular problem

"Do you feel the digital game has enhanced your understanding in basic programming concepts?"					
	Strongly Dislike	Dislike	Neutral	Like	Strongly Like
Number of responses	1	2	6	10	4
%	4.3	8.7	26.1	43.5	17.4
"Do you feel the digital game has enhanced your ability to conceptualize the solution to a particular problem?"					
	Strongly Dislike	Dislike	Neutral	Like	Strongly Like
Number of responses	0	1	6	14	2
%	0	4.3	26.1	60.9	8.7
"Do you feel the digital game is effective in helping you to first identify a particular pattern to problem solving and eventually implement is in a language that you are familiar with?"					
	Strongly Dislike	Dislike	Neutral	Like	Strongly Like
Number of responses	1	1	5	15	1
%	4.3	4.3	21.7	65.2	4.3

To summarize the study, the exploration of the digital game gave meaningful effects on participants' thoughts on programming. They feel the digital game has enhanced their understanding in basic programming concepts and ability to conceptualize the solution to a particular problem.

Most of them maintained their initial belief that it is good to learn programming before and after workshop. The limitation of the study is the duration period for the workshop. It is suggested to give longer time to explore the game so that students or participants able to get clearer picture of digital game and programming. The participants' dislike features also should be improved to increase the effectiveness of digital game in understanding introductory programming concepts.

Table 8. Second semester participants' opinions on effects of digital game in enhancing understanding in basic programming concepts and conceptualize the solution to a particular problem

"Do you feel the digital game has enhanced your understanding in basic programming concepts?"					
	Strongly Dislike	**Dislike**	**Neutral**	**Like**	**Strongly Like**
Number of responses	1	1	6	9	2
%	5.2	5.2	31.6	47.4	10.5
"Do you feel the digital game has enhanced your ability to conceptualize the solution to a particular problem?"					
	Strongly Dislike	**Dislike**	**Neutral**	**Like**	**Strongly Like**
Number of responses	0	1	4	10	4
%	0	5.2	21.1	52.6	21.1
"Do you feel the digital game is effective in helping you to first identify a particular pattern to problem solving and eventually implement is in a language that you are familiar with?"					
	Strongly Dislike	**Dislike**	**Neutral**	**Like**	**Strongly Like**
Number of responses	0	1	6	8	4
%	0	5.2	31.6	42.1	21.1

Table 9. Mean of likeliness for effectiveness elements of understanding introductory programming concepts

	First semester participants	Second semester participants
Elements	Mean of likeliness	Mean of likeliness
Playability	3.217	3.684
Multimedia adequacy	3.435	3.421
Sensory immersion	3.130	2.700
Ease of use	3.261	3.000
Challenge	3.522	3.526
Enjoyment	3.217	3.263
Learnability	3.609	3.580
Interaction	3.435	3.474
Context adequacy	3.304	3.263
Pedagogical utility	3.261	3.211

Table 10. Responses of first semester participant on their perception of programming after workshop

"Has the digital game helped to improve your perception of programming?"			
	Yes	**No**	**Not Sure**
Number of responses	13	2	8
%	56.5	8.7	34.8
"Has the digital game helped to dispel any negative thoughts you have on programming?"			
	Yes	**No**	**Not Sure**
Number of responses	15	6	2
%	65.2	26.1	8.7
"Would you consider learning programming in your spare time?"			
	Yes	**No**	**Not Sure**
Number of responses	20	3	0
%	87	13	0
"Do you believe it is good to learn programming?"			
	Yes	**No**	**Not Sure**
Number of responses	20	1	2
%	87	4.3	8.7

Table 11. Responses of second semester participant on their perception of programming after workshop

"Has the digital game helped to improve your perception of programming?"			
	Yes	**No**	**Not Sure**
Number of responses	13	4	2
%	68.4	21.1	10.5
"Has the digital game helped to dispel any negative thoughts you have on programming?"			
	Yes	**No**	**Not Sure**

Number of responses	10	3	6
%	52.6	15.8	31.6
"Would you consider learning programming in your spare time?"			
	Yes	**No**	**Not Sure**
Number of responses	13	1	5
%	68.4	5.3	26.3
"Do you believe it is good to learn programming?"			
	Yes	**No**	**Not Sure**
Number of responses	15	0	4
%	78.9	0	21.1

5. Conclusion

Programming concepts always appeared to be a challenging course to be learned, especially for novice students. Since digital game has its potential to boost the motivation of students and enable them to develop their knowledge efficiently, this study focuses on the quality of a digital game for learning programming concepts. The exploratory study indicated that participants felt that digital game has enhanced their understanding in basic programming concepts (60.9% of first semester students and 57.9% of second semester students) and the capability to conceptualize the solution to a particular problem (69.6% of first semester students and 73.7% of second semester students). These positive findings has shown that digital games is an effective tool which has helped to improve their perceptions on programming subjects and dispel their negative thoughts on programming subjects.

6. Acknowledgement

7. References

[1] V. Barr and C. Stephenson, "Bringing computational thinking to K-12: What is involved, what is the role of the computer science education community?", ACM Inroads, vol. 2 (1), pp. 48-54, 2011.

[2] A. Basawapatna, K.H. Koh and A. Repenning, "Using scalable game design to teach computer science from middle school to graduate school", Proceedings ITICSE '10, ACM Press, Bilkent, Ankara, Turkey, 2010.

[3] J. Denner, S. Bean and J. Martinez, "The girl game company: Engaging Latina girls in information technology", Afterschool Matter, vol. 8 (Spring 2009), pp. 26-35, 2009.

[4] J. Denner, "The girls creating games program: An innovative approach to integrating technology into middle school", Meridian: A Middle School Computer Technologies Journal, vol. 1(10), 2007.

[5] J. Denner, "What predicts middle school girls' interest in IT?", Integrated Journal of Gender in Science, Engineering and Technology, vol. 3(1), 2011.

[6] C.K. Chang, "Effects of Using Alice and Scratch in an Introductory Programming Course for Corrective Instruction", Journal of Educational Computing Research, vol. 51(2), pp.185-204, 2014.

[7] P. Gruenbaum, "Undergraduates Teach Game Programming Using Scratch", Computer, vol. 2, pp. 82-84, 2014.

[8] F. Kalelioğlu, "A new way of teaching programming skills to K-12 students: Code.org", Computers in Human Behavior, vol. 52, pp. 200-210, 2015.

[9] M.J. Lee, F. Bahmani, I. Kwan, J. LaFerte, P. Charters, A. Horvath, and A.J. Ko, "Principles of a debugging-first puzzle game for computing education", In Visual Languages and Human-Centric Computing (VL/HCC), 2014 IEEE Symposium on (pp. 57-64). IEEE, 2014.

[10] M. Akcaoglu, and M.J. Koehler, "Cognitive outcomes from the Game-Design and Learning (GDL) after-school program", Computers & Education, vol. 75, pp. 72-81, 2014.

[11] E. Núñez Castellar, A.All, L. de Marez, andJ. Van Looy, "Cognitive abilities, digital games and arithmetic performance enhancement", Computers & Education, vol. 85(C), pp. 123-133, 2015.

[12] Y.M. Cheng, S.H. Kuo, S.J. Lou, and R.C. Shih, "The Construction of an Online Competitive Game-Based Learning System for Junior High School Students", Turkish Online Journal of Educational Technology-TOJET,vol. 11(2), pp. 214-227, 2012.

[13] L.I. Dobrescu, B. Greiner, and A. "Motta, Learning economics concepts through game-play: An experiment",

International Journal of Educational Research, vol. 69, pp. 23-37. 2015.

[14] B. Manero, J. Torrente, A. Serrano, I. Martínez-Ortiz, and B. Fernández-Manjón, B., "Can educational video games increase high school students' interest in theatre?", Computers & Education, vol. 87, pp. 182-191, 2015.

[15] I. Ouahbi, F. Kaddari, H. Darhmaoui, A. Elachqar, andS. Lahmine, "Learning basic programming concepts by creating games with scratch programming environment", Procedia – Social and Behavioural Sciences, vol. 191, pp. 1479-1482, 2015.

[16] H.H.J. Mohd Nor, A.A. Azlan, I. Mohamad andM.N. Norsaniah, "Instructional design and learning theory on the development of C++ programming multimedia content", Procedia – Social and Behavioral Sciences, vol. 67, pp. 335-344, 2012.

[17] M.A. Ahmad Zamzuri, and M.D. Siti Rosminah, "Usability and user satisfaction of hardware-software interfacing visualization kit for novice learning programming", Procedia – Social and Behavioral Sciences, vol.103, pp. 1252-1260, 2013.

[18] Y. Wang, H. Li, Y. Feng, Y. Jiang, and Y. Liu, "Assessment of programming language learning based on peer code review model: implementation and experience report", Computers & Education, vol.59, pp.412-422, 2012.

[19] L. Major, An empirical investigation into the effectiveness of a robot simulator as a tool to support the learning of introductory programming, Dortor of Phylosophy Thesis, Keele University, 2014.